# Docker — Assignment 1

**Tasks To Be Performed:**  
1. Pull the Ubuntu container  
2. Run this container and map port 80 on the local  
3. Install Apache2 on this container  
4. Check if you are able to access the Apache page on your browser

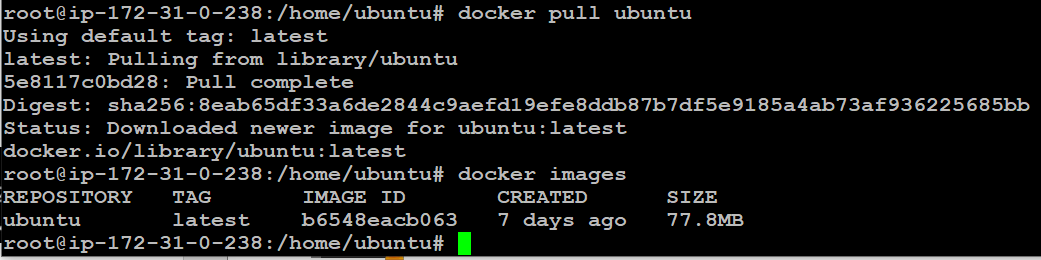
1. Pull the Ubuntu container

sudo apt update

sudo apt install docker.io

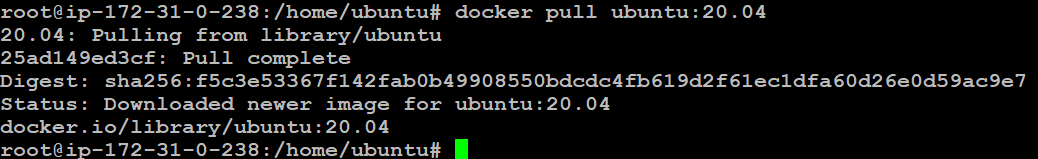
sudo docker pull ubuntu

sudo docker images (gives the list of docker images that are pulled)



To pull a specific version

sudo docker pull ubuntu:20.04



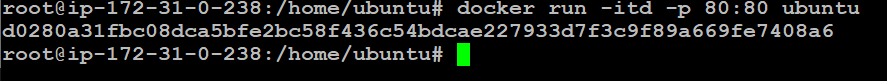
2. Run this container and map port 80 on the local

sudo docker run -itd -p 80:80 (Inteactes with the terminal in the background)

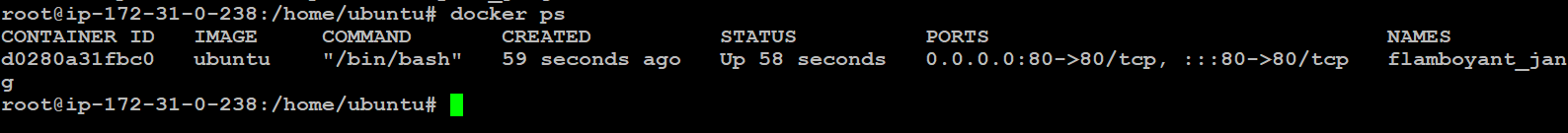
i — interactive

t - terminal

d- detached

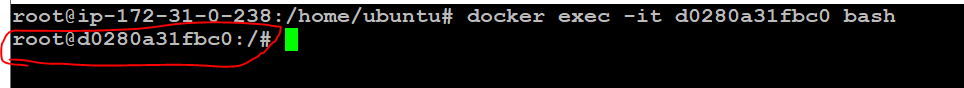


sudo docker ps (to check containers that are running)



3. Install Apache2 on this container

sudo docker exec -it <CONTINER ID> bash



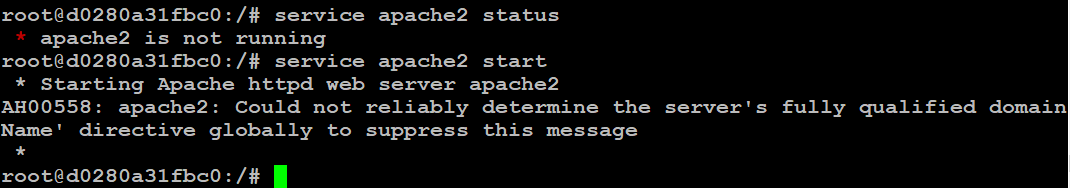
It will allow the docker container

apt update

apt install apache2 -y (To install the Apache server into the container)

service apache2 status (get the status of apache2 server)

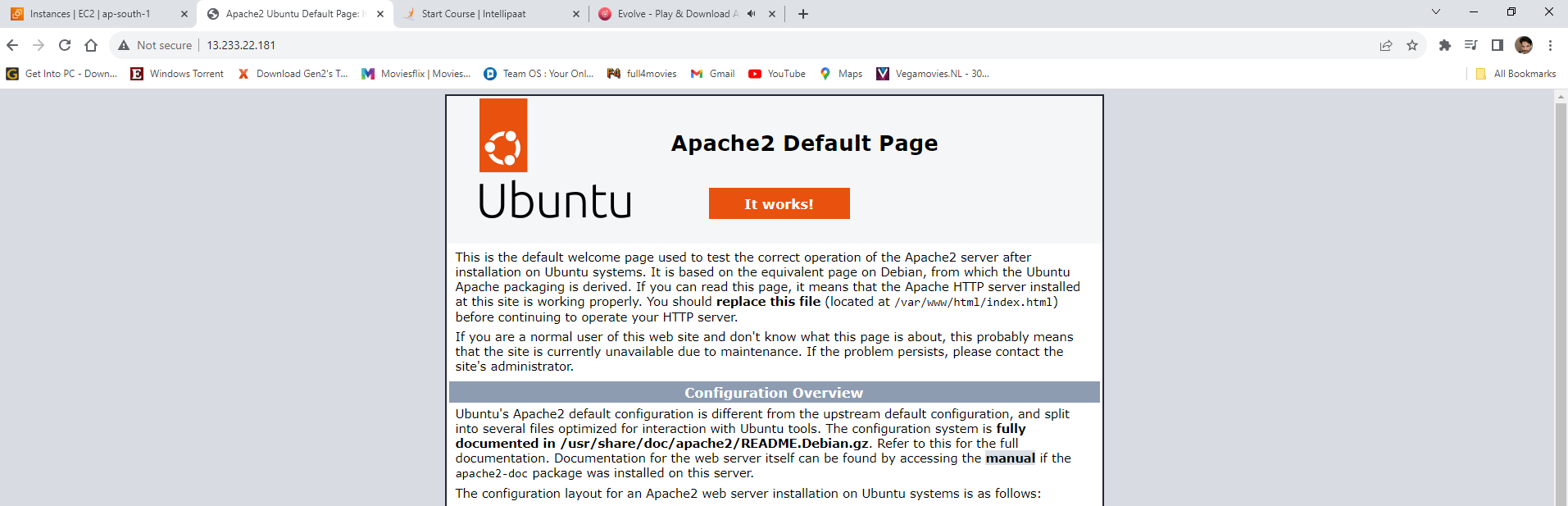
service apache2 start (Start Apache Server)

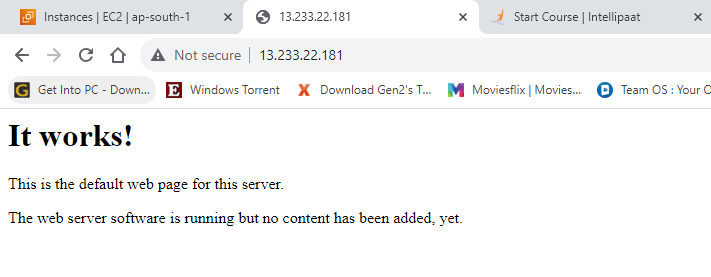


4. Check if you are able to access the Apache page on your browser

To check the page open Public IPv4 address:Port number

ex: http://13.233.22.181/:80





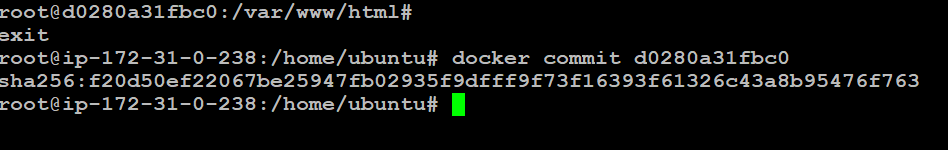
# Docker — Assignment 2

Tasks To Be Performed:  
1. Save the image created in assignment 1 as a Docker image  
2. Launch the container from this new image and map the port to 81  
3. Go inside the container and start the Apache2 service  
4. Check if you are able to access it on the browser

1. Save the image created in assignment 1 as a Docker image

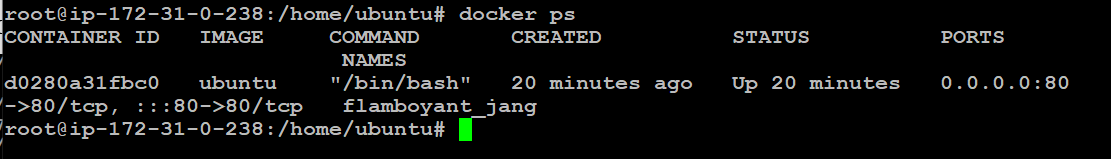
If I want to create the image with the same name we need to change the TAG. we can’t you the same TAG

sudo docker commit <CONTINER ID>



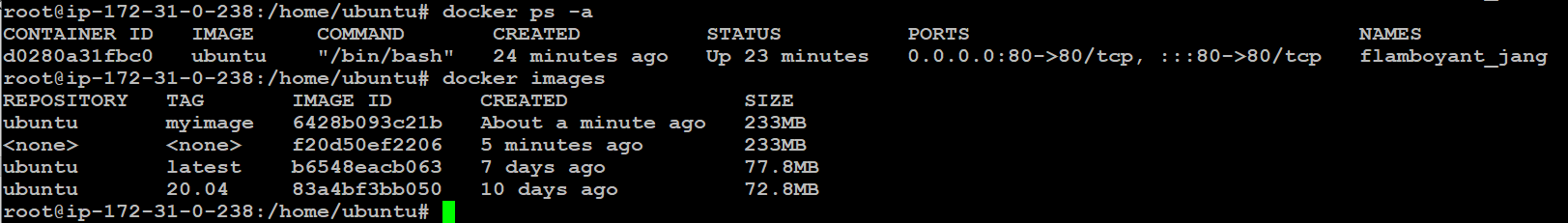
To get the container Id

sudo docker ps



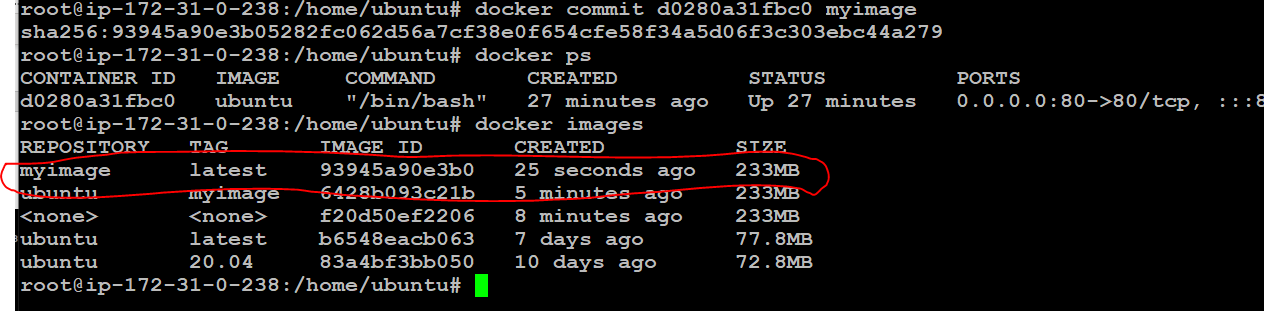
* Launch the container from this new image and map the port to 81

sudo docker commit <CONTINER ID> ubuntu:myimage

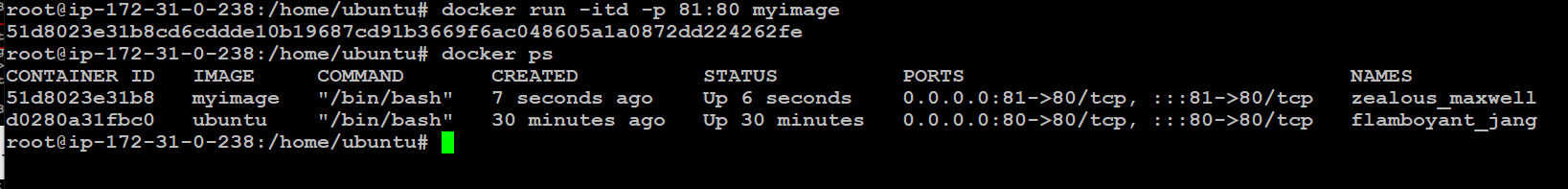


Actual assignment

sudo docker commit <CONTINER ID> myimage



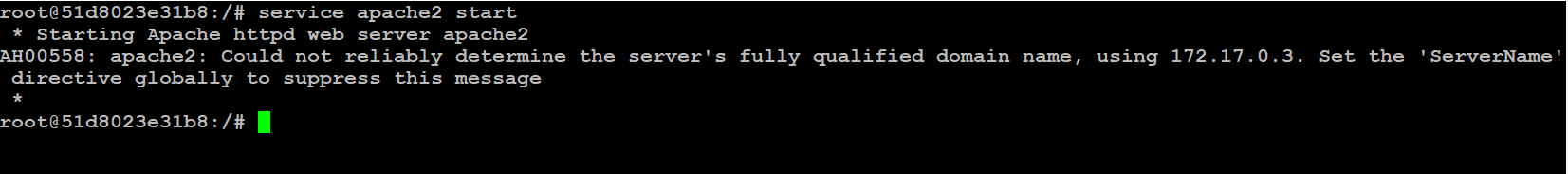
sudo docker run -itd -p 81:80 myimage



3.Go inside the container and start the Apache2 service

To get inside the container

sudo docker exec -it <CONTAINER ID> bash



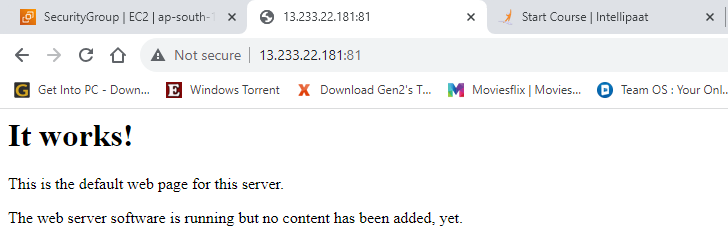
To check the server whether it is running or not



If not running then

service apache2 start (Command to start apache server)

4. Check if you are able to access it on the browser



# Docker — Assignment 3

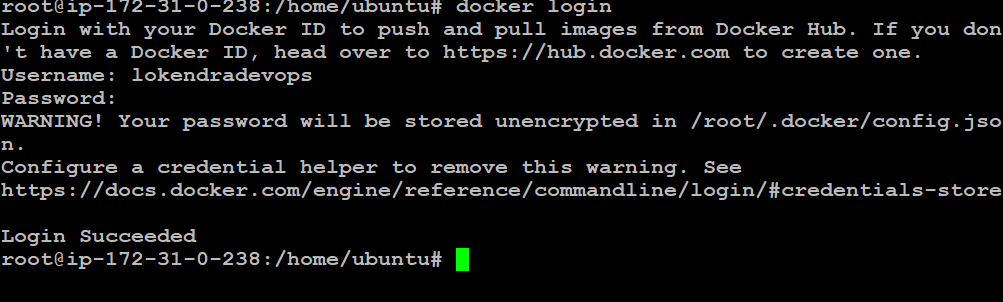
Tasks To Be Performed:  
1. Use the saved image in the previous assignment  
2. Upload this image on Docker Hub  
3. On a separate machine pull this Docker Hub image and launch it on port 80  
4. Start the Apache2 service  
5. Verify if you are able to see the Apache2 service

1. Use the saved image in the previous assignment

2. Upload this image on Docker Hub

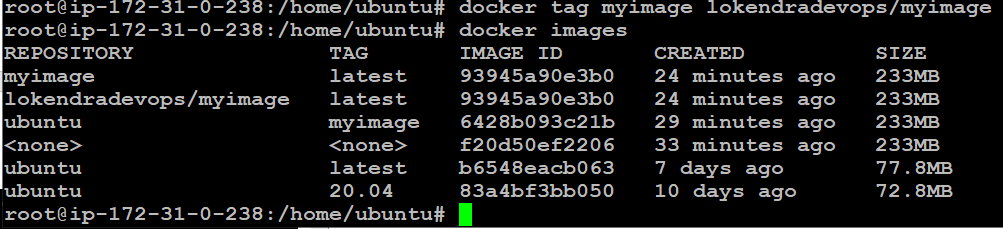
First Login into Docker using

sudo docker login

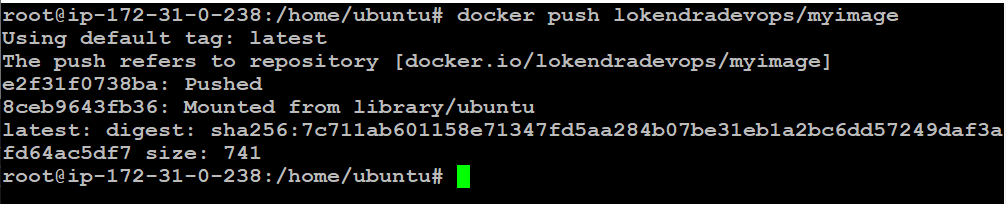


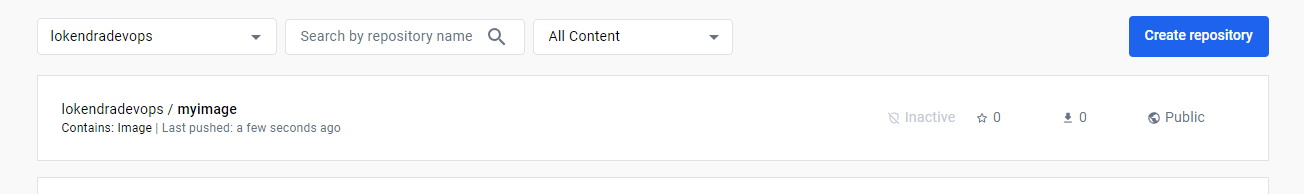
sudo docker tag myimage dockerhub\_id/myimage

sudo docker images



sudo docker push dockerhub\_id /myimage (It is used for pushing docker Image to Docker Hub)





3. On a separate machine pull this Docker Hub image and launch it on port 80

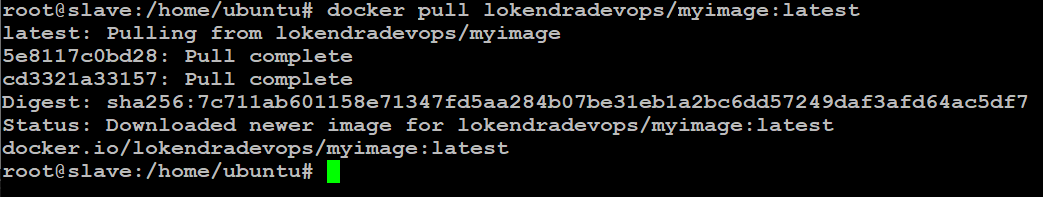
**\*In my case I have using 81 port because some issue with 81 port\***

create a new Instance

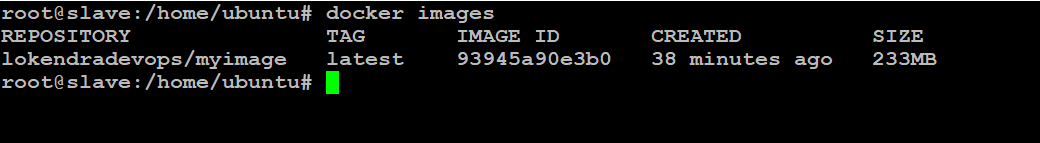
sudo apt update

sudo apt install docker.io

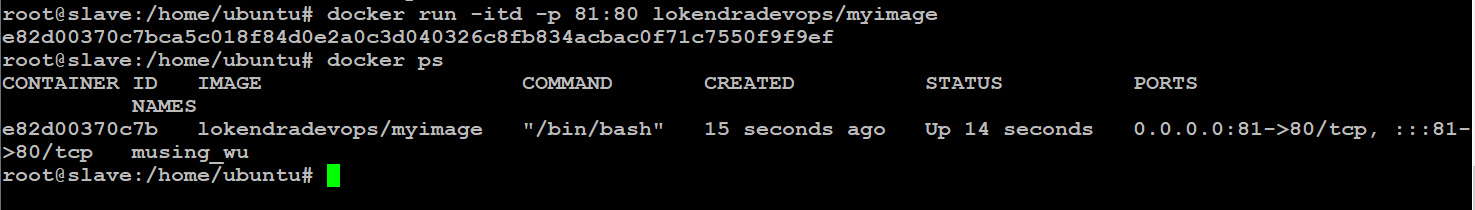
sudo docker pull dockerhub\_id /myimage:latest



sudo docker images



To create a container from the image that we pulled from docker hub

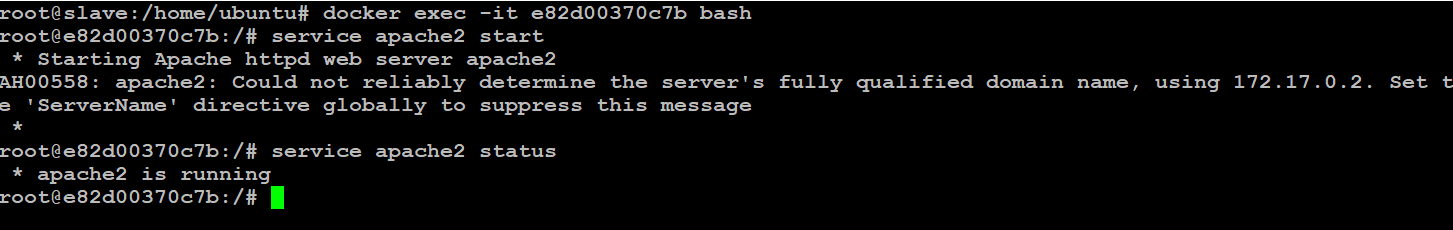


sudo docker exec -it container\_id bash

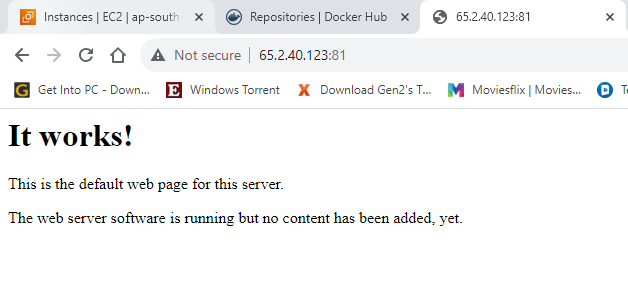


5. Verify if you are able to see the Apache2 service

check whether the server is running or not



And final check



**Docker — Assignment 4**

1. Create a Dockerfile with the following specs:  
   ● Ubuntu container  
   ● Apache2 installed  
   ● Apache2 should automatically run once the container starts
2. Create a Dockerfile with the following specs:  
   ● Ubuntu container  
   ● Apache2 installed  
   ● Apache2 should automatically run once the container starts

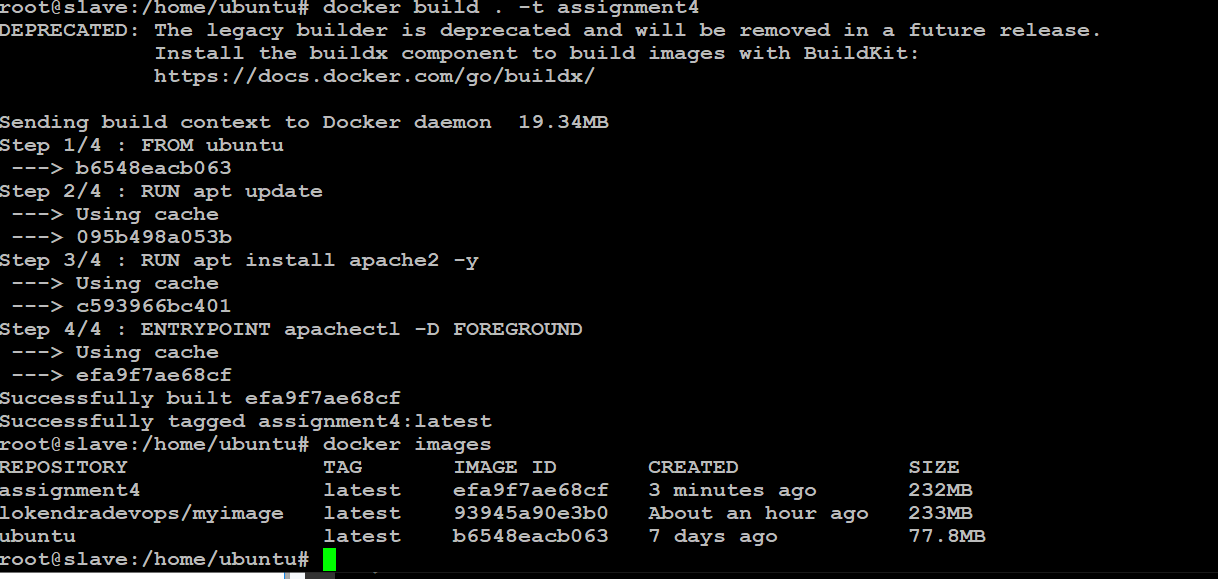
create a docker file

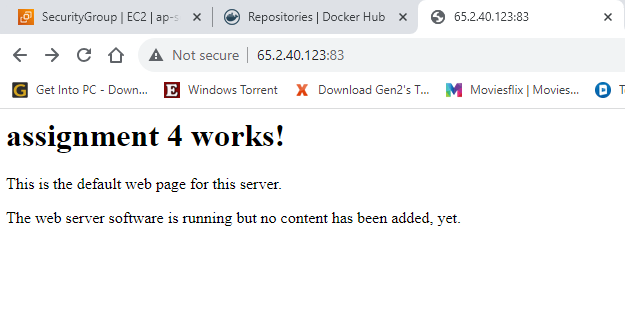
sudo nano Dockerfile

(ENTRYPOINT It set the priority as high)

FROM ubuntu  
RUN apt update  
RUN apt install apache2 -y  
ENTRYPOINT apachectl -D FOREGROUND

(ENTRYPOINT It set the priority as high)





# Docker — Assignment 5

**Tasks To Be Performed:**1. Create a sample HTML file  
2. Use the Dockerfile from the previous task  
3. Replace this sample HTML file inside the Docker container with the default page

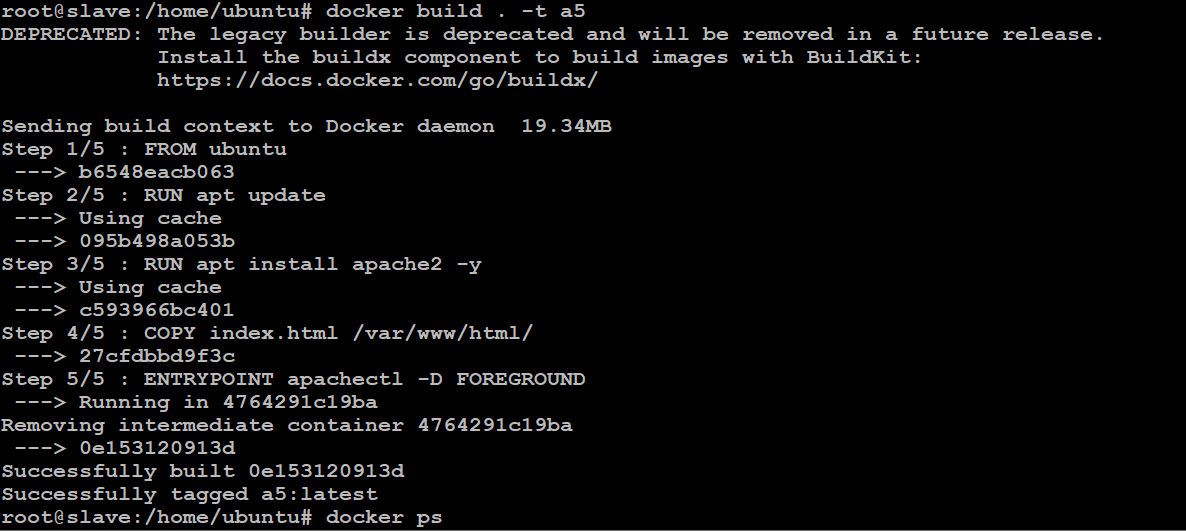
1. Create a sample HTML file  
2. Use the Dockerfile from the previous task  
3. Replace this sample HTML file inside the Docker container with the default page

sudo nano index.html

sudo nano Dockerfile

FROM ubuntu  
RUN apt update  
RUN apt install apache2 -y  
COPY index.html /var/www/html/  
ENTRYPOINT apachectl -D FOREGROUND

sudo docker build . -t a5 (To Build the image)



sudo docker run -itd -p 84:80 a5



Final browse and check

